PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-055725

(43)Date of publication of application : 20.02.2002

(51)Int.Cl. G06F 1/00
G06F 11/00
G06F 12/14
G06F 13/10

(21)Application number : 2000-234658

(22)Date of filing : 02.08.2000

(71)Applicant : INTEGRATED TECHNOLOGY EXPRESS INC

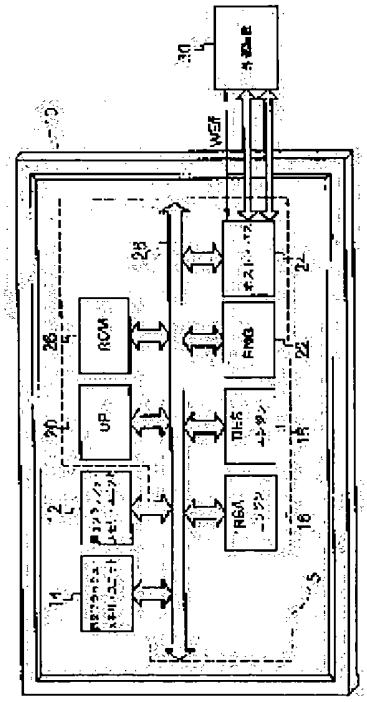
(72)Inventor : RIN EIMEI

(54) BIOS CHIP FOR MANAGING CODE AND METHOD FOR MANAGING THE SAME CODE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a code managing device for a BIOS and a BIOS chip capable of managing a code.

SOLUTION: A BIOS chip 10 is composed of a first flash memory unit 12 for preserving an inside BIOS, a second flash memory unit 14 for preserving code data, and an integrated code managing device 15 connected to an external device, the first flash memory, and the second flash memory. At the time of receiving a modifying command, the integrated code managing device generates code data, and transmits the code data to the second flash memory unit so that the code data can be preserved. Then, the encipherment of the coded data is executed. At last, the original coded data are compared with the decoded data, and only when the original coded data are coincident with the decoded data, the modification of the inside BIOS is permitted. The encipherment is made excutable by using asymmetrical RSA engines so that it is impossible to find out any correct RSA code. Therefore, it is possible to protect the inside BIOS data from modification due to the intrusion of any virus program.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)**公開特許公報**（Ａ）　(11)特許出願公開番号
特開2002－55725
（P2002－55725A）
(43)公開日　平成14年２月20日(2002.2.20)

(51)Int.Cl.⁷	識別記号	F.I	テーマコード（参考）
Ｇ０６Ｆ　1/00		Ｇ０６Ｆ　12/14	３１０Ａ　５Ｂ０１４
11/00		13/10	３３０Ｂ　５Ｂ０１７
12/14	３１０	9/06	６６０Ｊ　５Ｂ０７６
13/10	３３０		６６０Ｎ

審査請求　未請求　請求項の数７　ＯＬ　外国語出願　（全 17 頁）

(21)出願番号　特願2000－234658(P2000－234658)

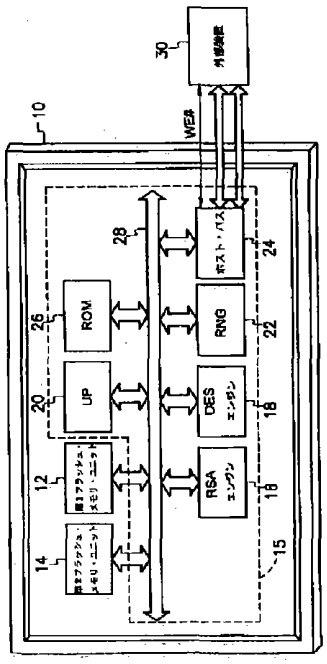
(22)出願日　平成12年８月２日(2000.8.2)

(71)出願人　500360998
聯陽半導體股▲ふん▼有限公司
台湾省新竹科學工業園区創新一路13号３樓
(72)発明者　林　永明
台湾台中縣烏日鄉三和村學田路４之９号
(74)代理人　100086368
弁理士　萩原　誠

Ｆターム(参考)　5B014 FB04
5B017 AA02 BA05 BA07 CA06
5B076 FA07 FD08

(54)【発明の名称】　コード管理ＢＩＯＳチップとそのコード管理方法

(57)【要約】　（修正有）
【解決手段】　ＢＩＯＳチップ１０は、内部ＢＩＯＳを保持する第１フラッシュ・メモリ・ユニット１２、コード・データを保持する第２フラッシュ・メモリ・ユニット１４および外部装置、第１フラッシュ・メモリおよび第２フラッシュ・メモリに接続している集積コード管理装置１５からなる。修正コマンドを受信すると、集積コード管理装置はコード・データを発生させ、それを第２フラッシュ・メモリ・ユニットに送信し、保存する。次にコード化データの暗号化が実行される。最後に、元のコード化データと復号化データは比較され、元のコード化データと復号化データが、互いに一致する場合にのみ、内部ＢＩＯＳデータの修正が許可される。
【効果】　暗号化は、非対称性のＲＳＡエンジンを使用することにより実行可能なので、正しいＲＳＡコードが見出されることは不可能である。したがって、内部ＢＩＯＳデータを、ウイルス・プログラムの侵入による修正から守ることができる。



【特許請求の範囲】
【請求項１】　外部デバイスからの入力修正コマンドを検証した後に、内部ＢＩＯＳデータを修正する許可を与えるコード管理装置を内部に有するＢＩＯＳチップであって、
前記内部ＢＩＯＳデータを保持する第１フラッシュ・メモリ・ユニットと、
セキュリティ・データを保持する第２フラッシュ・メモリ・ユニットと、
前記外部装置、前記第１フラッシュ・メモリ・ユニットおよび前記第２フラッシュ・メモリ・ユニットに接続された集積コード管理装置とからなり、
前記集積コード管理装置は、前記修正コマンドを受信し、前記セキュリティ・データを前記第２フラッシュ・メモリ・ユニットに保存するために送信し、前記セキュリティ・データを暗号化して暗号化データを発生させ、前記暗号化データを復号化するために前記外部装置に送信し、
前記外部装置での復号化後、前記復号化データは、前記元のセキュリティ・データと比較するために前記集積コード管理装置へ返信され、前記セキュリティデータと一致する場合に、前記内部ＢＩＯＳデータが、前記外部装置から提供されるデータに置き換えられることを特徴とするＢＩＯＳチップ。
【請求項２】　請求項１に記載のＢＩＯＳチップにおいて、
前記集積コード管理装置がさらに、
前記外部装置に接続されて、前記ＢＩＯＳに対する前記修正コマンドおよびデータを受信するホスト・バスと、
前記ホスト・バスに接続されて、前記修正コマンドを受信し、検証要求コマンドを送信するマイクロコントローラと、
前記マイクロコントローラに接続されて、前記検証要求コマンドを受信し、第１乱数を発生させる乱数発生器と、
前記乱数発生器および前記ホスト・バスに接続されて、前記第１乱数を受信してＲＳＡ暗号化データを発生させるＲＳＡエンジンとからなり、
前記ＲＳＡ暗号化データは、前記外部装置へ送信され、前記外部装置は前記暗号化データを復号化して第２乱数を発生させ、
前記第２乱数は返信されて、第１乱数と比較され、前記第１および第２乱数が一致する時にのみ、前記内部ＢＩＯＳデータが外部データに置き換えられることを特徴とするＢＩＯＳチップ。
【請求項３】　請求項１に記載のＢＩＯＳチップにおいて、
前記集積コード管理装置がさらに、
前記外部装置に接続されて、前記ＢＩＯＳに対する前記修正コマンドおよびデータを受信するホスト・バスと、
前記ホスト・バスに接続されて、前記修正コマンドを受信し、検証要求コマンドを送信するマイクロコントローラと、
前記マイクロコントローラに接続されて、前記検証要求コマンドを受信し、第１乱数を発生させる乱数発生器と、
前記乱数発生器および前記ホスト・バスに接続されて、前記第１乱数を受信してＤＥＳ暗号化データを発生させるデータ暗号化規格（ＤＥＳ）エンジンとからなり、
前記ＤＥＳ暗号化データは、前記外部装置へ送信され、前記外部装置は前記暗号化データを復号化して第２乱数を発生させ、
前記第２乱数は返信されて、第１乱数と比較され、前記第１および第２乱数が一致する時にのみ、前記内部ＢＩＯＳデータが外部データに置き換えられることを特徴とするＢＩＯＳチップ。
【請求項４】　外部装置からの内部ＢＩＯＳデータ修正要求が妥当であるかどうかをチェックした後に、実際に修正許可を与えるコード管理方法であって、
ＢＩＯＳを修正するためのデータをレジスタに保存するステップと、
ランダム・コードを発生させ、前記ランダム・コードを蓄積するステップと、
前記ランダム・コードを暗号化して、暗号化データを発生させるステップと、
前記暗号化データを前記外部装置へ送信するステップと、
前記暗号化データを前記外部装置において復号化するステップと、
前記元のランダム・コードを前記復号化データと比較して、一致するかどうかをチェックするステップと、
前記復号化データと前記ランダム・コードとが一致する場合に、前記内部ＢＩＯＳデータの修正を許可するステップと、
前記復号化データと前記ランダム・コードとが一致しない場合に、前記内部ＢＩＯＳデータのいかなる修正をも許可しないステップとからなることを特徴とするコード管理方法。
【請求項５】　請求項４に記載の方法において、
前記ランダム・コードを発生させるステップにおいて、乱数発生器を使用することを特徴とする方法。
【請求項６】　請求項４に記載の方法において、
前記ランダム・コードを暗号化して、暗号化データを発生させるステップにおいて、ＲＳＡエンジンを使用することを特徴とする方法。
【請求項７】　請求項４に記載の方法において、
前記ランダム・コードを暗号化して、暗号化データを発生させるステップにおいて、データ暗号化規格（ＤＥＳ）エンジンを使用することを特徴とする方法。

【発明の詳細な説明】

【０００１】
【発明の属する技術分野】本発明は、基本入出力システム（ＢＩＯＳ）に関し、特に、ＢＩＯＳ内部のコード管理に関する。
【０００２】
【従来の技術】一般に、パーソナル・コンピュータの電源を入れると、パーソナル・コンピュータ内部の中央処理ユニット（ＣＰＵ）が自動的に一連のコマンドを実行するが、これらのコマンドが実行する動作は、下記の３種類に大別できる。すなわち、
１．システム構成分析を実行するコマンドであって、このシステム構成分析では、そのコンピュータ・システムのＣＰＵのタイプ、記憶容量、ソフトウェアおよびハードウェアのタイプおよび数量、浮動小数点計算装置などに関する情報が決定される。このような情報は、その後の動作に参照される。
２．自己テストの起動（ＰＯＳＴ）を実行するコマンドであって、このＰＯＳＴでは、メモリ・ユニット、チップセット、ＣＭＯＳ、保存データ、キーボード、磁気ディスク装置などのハードウェアの状態をチェックして、もしもエラーがあれば、それらをユーザに知らせる。
３．オペレーティング・システムをダウンロードするコマンドであって、例えば、ブートストラップ・ローダと呼ばれる小さなプログラムにより、オペレーティング・システム（例えばＭＳ　ＤＯＳあるいはＷｉｎｄｏｗｓ（登録商標）９５/９８）がハードディスクから探し出され、ダウンロードされる。その後、オペレーティング・システムに制御権が移譲され、起動段階が終了する。
【０００３】コンピュータ・システムを起動させるこのようなコマンドのすべては、よく基本入出力システム（ＢＩＯＳ）プログラムと呼ばれている。つまりＢＩＯＳプログラムは、第１の実行プログラムとみなすことができる。もし起動過程が途中で止まってしまった場合には、たいてい、何らかの問題がハードウェアのどこかに発生しているので、コンピュータを正常に動作させるためには、これらのハードウェアの問題を解決する必要がある。
【０００４】
【発明が解決しようとする課題】ＢＩＯＳプログラムは通常、フラッシュＲＯＭチップに保存されているが、このフラッシュＲＯＭは、内部データへのランダム・アクセスおよび内部データの修正ができるタイプのメモリである。フラッシュＲＯＭチップ内のデータは、電源をオフにしても保持できるので、フラッシュＲＯＭチップは、パーソナル・コンピュータの起動プログラムを保存するのに使用できる。しかしながら、プログラム上の特性により、フラッシュＲＯＭはまた、ウイルス・プログラムの攻撃に弱く、修復不能なシステム・ダメージを被る可能性がある。
【０００５】したがって、本発明の第１の目的は、ＢＩＯＳ用のコード管理装置を提供することである。集積コード管理装置を使用して、コード化データを発生させ、保存する。非対称性のＲＳＡエンジンあるいは対称性のデータ暗号化規格（ＤＥＳ）エンジンにより、暗号化処理が実行される。暗号化データは、外部装置へ送信され、復号化される。そして最後に、復号化データは返信され、暗号化データと比較される。前記比較において、暗号化データとの一致が確認されて初めて、ＢＩＯＳデータの修正が許可される。本発明の第２の目的は、コード管理のできるＢＩＯＳチップを提供することである。このＢＩＯＳチップは、外部装置からのＢＩＯＳ修正コマンドの妥当性を評価した後に、実際のデータ修正を許可する。
【０００６】
【課題を解決するための手段】本発明の目的に従って、上記のおよびその他の長所を達成するために、ここで具体化して大まかに説明するように、本発明は、コード管理装置を有するＢＩＯＳチップを提供する。このＢＩＯＳチップは、第１フラッシュ・メモリ・ユニット、第２フラッシュ・メモリ・ユニットおよび集積コード管理装置からなる。第１フラッシュ・メモリ・ユニットは、内部ＢＩＯＳデータを保持する装置である。第２フラッシュ・メモリ・ユニットは、暗号化データを保持する装置である。集積コード管理装置は、外部装置、第１フラッシュ・メモリおよび第２フラッシュ・メモリに接続されている。修正コマンドを受信すると、集積コード管理装置はコード・データを発生させる。コード化データの暗号化が、次に実行される。暗号化データは、外部装置へ送信されて、復号化される。最後に、元の暗号化データと復号化データが一致するかどうか比較される。元の暗号化データと復号化データが互いに一致する場合にのみ、内部ＢＩＯＳデータの修正が許可される。
【０００７】前記コード管理装置は、ホスト・バス、マイクロコントローラ、乱数発生器およびＲＳＡエンジンあるいはデータ暗号化規格（ＤＥＳ）エンジンからなる。ホスト・バスは、外部装置に接続されていて、ＢＩＯＳに対する修正コマンドおよび修正データを受信する。マイクロコントローラは、ホスト・バスに接続されていて、修正コマンドを受信し、その後、検証要求コマンドを送信する。乱数発生器は、マイクロコントローラに接続されていて、検証要求コマンドを受信し、その後、第１乱数を発生させる。ＲＳＡエンジンあるいはＤＥＳエンジンは、乱数発生器およびホスト・バスに接続されていて、乱数発生器から第１乱数を受信し、その後、ＲＳＡ暗号化データあるいはＤＥＳ暗号化データを発生させる。暗号化データは、ホスト・バスを介して外部装置へ送信される。外部装置は、暗号化データを復号化し、第２乱数を発生させる。最後に第２乱数は、第１乱数と一致するかどうかをチェックするために第１乱数

と比較され、一致する場合にのみ、内部ＢＩＯＳデータの修正が許可される。

【０００８】本発明はまた、外部装置からの内部ＢＩＯＳデータ修正要求が妥当であるかどうかをチェックした後、実際の修正が許可されるコード管理方法を提供する。最初、ＢＩＯＳを修正するデータはレジスタに保存されている。ランダム・コードが発生し、保存される。それからランダム・コードは、暗号化されて、暗号化データが発生する。暗号化データは、外部装置に送信され、復号化される。復号化データは、元のランダム・コードと比較されて、一致するかどうかチェックされる。復号化データとランダム・コードが一致する場合には、ＢＩＯＳデータを修正する許可が与えられる。比較の結果、一致しない場合には、ＢＩＯＳデータの修正は許可されない。

【０００９】ランダム・コードは乱数発生器により発生させることができ、非対称性のＲＳＡエンジンあるいは対称性のデータ暗号化規格（ＤＥＳ）エンジンによって暗号化することができる。以上の大まかな説明および以下の詳細な説明は、どちらも例示的なもので、特許請求の範囲において、本発明がさらに詳しく説明されることを理解されたい。

【００１０】

【発明の実施の形態】以下、この発明にかかる好適な実施例を図面に基づいて説明する。同一あるいは同種の部分を説明するために、図面および説明には、可能な限り同じ参照番号を使用する。一般に、ＢＩＯＳプログラムはフラッシュＲＯＭチップに保存されている。フラッシュＲＯＭチップは、外部コマンドによってプログラム可能なので、ウイルス・プログラムによる攻撃の格好のターゲットとなり、システムが修復困難なくらいの機能不全に陥ってしまう。ＢＩＯＳプログラムが不正に書き換えられるのを防止するために、コード・データを発生させて、このデータを非対称性のＲＳＡエンジンあるいはデータ暗号化規格（ＤＥＳ）エンジンを用いて暗号化する。暗号化データは、ＢＩＯＳデータ修正を要求するどんな外部装置へでも送信される。暗号化データは、外部装置により復号化される。復号化データは返信され、コード・データと比較される。復号化データとコード・データが互いに一致する場合にのみ、内部ＢＩＯＳデータを修正する許可が与えられる。その結果、ＢＩＯＳチップ内部のＢＩＯＳデータが、高水準の安全性を有することができる。

【００１１】図１は、本発明の１つの好適な実施形態による、ＢＩＯＳチップ内部のコード管理システムを示す概略図である。図１のように、コード管理ＢＩＯＳチップ１０は、第１フラッシュ・メモリ・ユニット１２、第２フラッシュ・メモリ・ユニット１４および集積コード管理装置１５からなる。集積コード管理装置１５はさらに、ＲＳＡエンジン１６、ＤＥＳエンジン１８、マイクロコントローラ２０、ランダム・コード発生器２２、ホスト・バス２４、ＲＯＭユニット２６および内部バス２８からなる。第１フラッシュ・メモリ・ユニット１２は、内部ＢＩＯＳデータを保存し、第２フラッシュ・メモリ・ユニット１４は、暗号化データを保存する。まず初めに、外部装置３０から修正コマンドＷＥ＃がＢＩＯＳチップ１０へ送信され、修正コマンドＷＥ＃が、ＢＩＯＳチップ内にあるコード管理装置により審査される。外部装置３０からの、ＢＩＯＳを修正するいかなるデータも、審査を通過した時のみ受信される。

【００１２】ホスト・バス２４を介して修正コマンドＷＥ＃を受信すると、ＢＩＯＳチップ１０内部のマイクロコントローラ２０は、修正データを内部レジスタ（図示せず）に保存するように動作する。その間に、マイクロコントローラ２０はまた、ランダム・コード発生器２２を作動させ、ランダム・コードＮを発生させる。ランダム・コードＮは、内部バス２８を介して第２フラッシュ・メモリ１４に送信されて、保存される。同時に、ランダム・コードＮは、ＲＳＡエンジン１６によって暗号化され、暗号化データＲＳＡ（Ｎ）が発生する。あるいはランダム・コードＮは、ＤＥＳエンジン１８によって暗号化されると、暗号化データＤＥＳ（Ｎ）が発生する。さらに、データ暗号化の信頼度を向上させるために、ＲＳＡエンジン１６およびＤＥＳエンジン１８を併用してもよい。

【００１３】暗号化データＲＳＡ（Ｎ）あるいはＤＥＳ（Ｎ）は、内部バス２８およびホスト・バス２４を介して外部装置３０に返信される。次に、外部装置３０は、暗号化データＲＳＡ（Ｎ）あるいはＤＥＳ（Ｎ）を復号化して、復号化データＮを発生させる。ホスト・バス２４および内部バス２８を介して、復号化データＮは、再びＢＩＯＳチップ１０へ送信される。復号化データＮは、第２フラッシュ・メモリ１４に保存されているランダム・コードＮと比較される。復号化データＮとランダム・コードＮが一致する場合には、ＢＩＯＳを修正するデータが外部装置３０から入力され、修正コマンドＷＥ＃が消える。逆に、復号化データＮとランダム・コードＮが一致しない場合には、外部装置３０からのいかなる修正データも拒絶される。

【００１４】ＲＳＡエンジン16によりコード・データを暗号化して、暗号化データＲＳＡ（Ｎ）を発生させること、あるいはＤＥＳエンジン１８によりコード・データを暗号化して、暗号化データＤＥＳ（Ｎ）を発生させることによって、ウィルス・プログラムは正確なＲＳＡあるいはＤＥＳコードを獲得することが不可能となる。したがって、ウイルス・プログラムが誤ったデータをいくらＢＩＯＳに送り込もうとしても、必ず失敗する。さらに、暗号化データＲＳＡ（Ｎ）あるいはＤＥＳ（Ｎ）を復号化するために外部装置へ送信し、次いで復号化データを審査のためにコード管理装置へ再送信すること

で、内部ＢＩＯＳチップ・データに対する別の安全性が高まる。

【００１５】以上のごとく、この発明を好適な実施例により開示したが、もとより、この発明を限定するためのものではなく、当業者であれば容易に理解できるように、この発明の技術思想の範囲内において、適当な変更ならびに修正が当然なされうるものであるから、その特許権保護の範囲は、特許請求の範囲および、それと均等な領域を基準として定めなければならない。

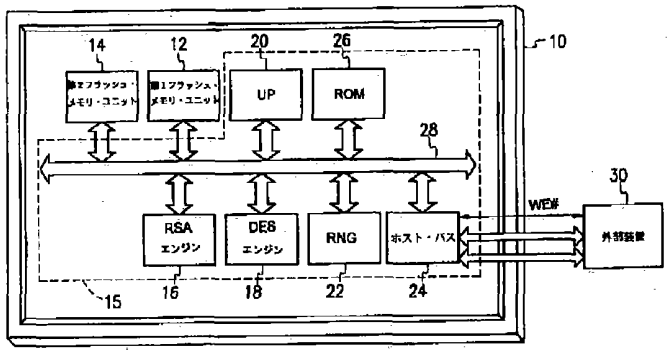
【図面の簡単な説明】

【図１】本発明の１つの好適な実施形態による、ＢＩＯＳチップ内部のコード管理システムを示す概略図。

【符号の説明】

１０：　コード管理ＢＩＯＳチップ
１２：　第１フラッシュ・メモリ・ユニット
１４：　第２フラッシュ・メモリ・ユニット
１５：　集積コード管理装置
１６：　ＲＳＡエンジン
１８：　ＤＥＳエンジン
２０：　マイクロコントローラ
２２：　ランダム・コード発生器
２４：　ホスト・バス
２６：　ＲＯＭユニット
２８：　内部バス
３０：　外部装置
ＷＥ＃　修正コマンド

【図１】



【外国語明細書】

1. TITLE OF THE ENVENTION

CODE ADMINISTRATION OF BIOS

2. WHAT IS CLAIMED IS:

1. A BIOS chip having a code administrator therein for inspecting input modification command from an external device before granting a permission to modify internal BIOS data, comprising:

a first flash memory unit for holding internal BIOS data;

a second flash memory unit for holding security data; and

an integrated code administration device connected to the external device, the first flash memory unit and the second flash memory unit, wherein the integrated code administration device receives the modification command, transmits the security data to the second flash memory unit for storage, encrypts the security data to produce encrypted data and sends the encrypted data to the external device for decryption, after decryption in the external device, the decrypted data is returned to the integrated code administration device to compare with the original security data, if there is a match between the said data, internal BIOS data is replaced by the data provided by the external device.

2. The BIOS chip of claim 1, wherein the integrated code administration device further includes:

a host bus connected to the external device for receiving the modification command and modification data for the BIOS;

a micro-controller connected to the host bus for receiving the modification command and sending out a request-for-inspection command;

a random number generator connected to the micro-controller for receiving the request-for-inspection command and generating a first random number; and

a RSA engine, wherein the RSA engine is connected to the random number generator and the host bus for receiving the first random number to produce RSA encrypted data, the RSA encrypted data are sent to the external device, the external device decrypts the encrypted data to produce a second random number, the second random number is returned and compared with the first random number, internal BIOS data is replaced by external data only when the first and the second random number match.

3. The BIOS chip of claim 1, wherein the integrated code administration device further includes:

a host bus connected to the external device for receiving the modification command and modification data for the BIOS;

a micro-controller connected to the host bus for receiving the modification command and sending out a request-for-inspection command;

a random number generator connected to the micro-controller for receiving the request-for-inspection command and generating a first random number; and

a data encryption standard (DES) engine, wherein the DES engine is connected to the random number generator and the host bus for receiving the first random number to produce DES encrypted data, the DES encrypted data are sent to the external device,

the external device decrypts the encrypted data to produce a second random number, the second random number is returned and compared with the first random number, internal BIOS data is replaced by external data only when the first and the second random number match.

4. A code administration method for checking the validity of request from an external device for modifying the internal BIOS data before granting an actual permission for the modification, comprising the steps of:

storing data for modifying the BIOS in registers;

generating a random code and storing up the random code;

encrypting the random code to produce encrypted data;

sending the encrypted data to the external device;

decrypting the encrypted data in the external device;

comparing the original random code with the decrypted data to check for conformity;

granting the modification of internal BIOS data when there is a match between the decrypted data and the random code; and

rejecting any modification to internal BIOS data when there is a mismatch between the decrypted data and the random code.

5. The method of claim 4, wherein the step of producing the random code includes using a random number generator.

6. The method of claim 4, wherein the step of encrypting the random code to produce encrypted data includes using a RSA engine.

7. The method of claim 4, wherein the step of encrypting the random code to produce encrypted data includes using a data encryption standard (DES) engine.

3. DETAILED DESCRIPTION OF THE INVENTION

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a basic input/output system (BIOS). More particularly, the present invention relates to code administration inside a BIOS.

2. Description of the Related Art

In general, after a personal computer is turned on, the central processing unit (CPU) within the personal computer will carry out a sequence of commands automatically. The operations carried out by these commands can be roughly classified into three major types, namely:

1. Commands that perform a system configuration analysis. In the system configuration analysis, information regarding the CPU type, memory size, software and hardware type and quantities, any floating point computation device and so on of the computer system are determined. Such information can be used as a reference in subsequent actions.

2. Commands that perform a power on self test (POST). In the POST, hardware states of memory units, chipsets, CMOS, storage data, keyboard, magnetic disk machines are checked. If any errors are discovered, those errors are reported back to the user.

3. Commands that download an operating system. Through a small program known as a 'bootstrap loader', an operating system (such as MS DOS or Window 95/98)

is found and downloaded from a hard disk, for example. Thereafter, power of control is transferred to the operating system before the end of the start-up session.

All the said commands on starting a computer system are often referred to as a basic input/output system (BIOS) program. In short, the BIOS program can be regarded as the first program to be executed. If the start-up process is stuck, some hardware problems have probably occurred somewhere. To operate the computer successfully, these hardware problems must be removed.

The said BIOS program is generally stored in a flash ROM chip. Flash ROM is a type of memory that permits random access and modification of internal data. Since data within a flash ROM chip is retained after power off, the flash ROM chip can be used to store the start-up program of a personal computer. However, because of programmable characteristics, flash ROM is also vulnerable to attack by virus programs leading to possible irreversible system damages.

SUMMARY OF THE INVENTION

Accordingly, one object of the present invention is to provide a code administrator for BIOS. Using an integrated code administration device, coded data are generated and stored. A non-symmetrical RSA engine or a symmetrical data encryption standard (DES) engine is used to carry out code processing. Coded data are transmitted to an external device for decoding. Finally, the decoded data is returned and compared with the coded data. Any modification of BIOS data is permitted only

after a positive identification is shown in the said comparison.

A second object of this invention is to provide a BIOS chip having code administration capability. The BIOS chip is able to assess the validity of any BIOS modification command coming from an external device before granting any permission for actual modification of data.

To achieve these and other advantages and in accordance with the purpose of the invention, as embodied and broadly described herein, the invention provides a BIOS chip having a code administrator. The BIOS chip includes a first flash memory unit, a second flash memory unit and an integrated code administration device. The first flash memory unit is a device for holding some internal BIOS data. The second flash memory unit is a device for holding coded data. The integrated code administration device is connected to an external device, the first flash memory and the second flash memory. After receiving a modification command, the integrated code administration device generates code data. A data encryption of the coded data is next carried out. The encrypted data is sent to the external device for de-encryption. Finally, the original coded data and the decrypted data is compared for conformity. Only when the original coded data and the decrypted data match each other will the permission to modify any internal BIOS data be granted.

The said administration device includes a host bus, a micro-controller, a random number generator and a RSA engine or a data encryption standard (DES) engine. The host bus is connected to the external device for receiving any modification command and modification data for the BIOS. The micro-controller is connected to the host bus

for receiving any modification command and sending out a request-for-inspection command thereafter. The random number generator is connected to the micro-controller for receiving the request-for-inspection command and generating a first random number thereafter. The RSA engine or the DES engine is connected to the random number generator and the host bus for receiving the first random number and generating a RSA encrypted data or DES encrypted data thereafter. The encrypted data is sent to the external device via the host bus. The external device then decrypts the encrypted data to produce a second random number. Lastly, the second random number is compared with the first random number to check if they match. Only when the said comparison is a match will any modification of internal BIOS data be granted.

This invention also provides a code administration method for checking the validity of request from an external device for modifying the internal BIOS data before granting an actual permission for the modification. First, data for modifying the BIOS are stored in registers. A random code is generated and stored. The random code is then encrypted to produce encrypted data. The encrypted data is sent to the external device for decryption. The decrypted data is compared with the original random code to check for conformity. If there is a match in the decrypted data and the random code, permission for changing BIOS data is granted. If the said comparison does not result in a match, modification of BIOS data is rejected.

The random code can be generated through a random number generator. The random code can be encrypted by a non-symmetrical RSA engine or a symmetrical data encryption standard (DES) engine.

It is to be understood that both the foregoing general description and the following detailed description are exemplary, and are intended to provide further explanation of the invention as claimed.

DESCRIPTION OF THE PREFERRED EMBODIMENT

Reference will now be made in detail to the present preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers are used in the drawings and the description to refer to the same or like parts.

In general, BIOS programs are stored in a flash ROM chip. Since flash ROM chip can be programmed through external commands, ROM chip is an easy target for attack by virus programs leading to irreparable system mal-function. To prevent any illegal tampering with BIOS program, code data is generated and then the coded data is encrypted using a non-symmetrical RSA engine or a data encryption standard (DES) engine. The encrypted code data is transmitted to any external device requesting BIOS data modification. The encrypted data is decrypted by the external device. The decrypted data is transmitted back and compared with the code data. Only when the decrypted data and the code data conforms to each other will permission for changing internal BIOS data be granted. Consequently, BIOS data inside a BIOS chip can have a higher level of security.

Fig. 1 is a sketch showing the code administration system within a BIOS chip

according to one preferred embodiment of this invention. As shown in Fig. 1, the code administration BIOS chip 10 includes a first flash memory unit 12, a second flash memory unit 14 and an integrated code administration device 15. The integrated code administration device 15 further includes a RSA engine 16, a DES engine 18, a micro-controller 20, a random code generator 22, a host bus 24, a ROM unit 26 and an internal bus 28. First flash memory unit 12 is used to store internal BIOS data and second flash memory unit 14 is used to store encoded data.

First, when an external device 30 sends a modification command WE# to the BIOS chip 10, the modification command WE# is examined by the code administrator inside the BIOS chip. Any data from the external device 30 for modifying the BIOS is accepted only on passing the examination.

On receiving the modification command WE# via the host bus 24, micro-controller 20 inside BIOS chip 10 operates to store the modification data in an internal register (not shown in the figure). In the meantime, micro-controller 20 also activates random code generator 22 to produce a random code N. The random code N is transmitted to second flash memory 14 for storage via internal bus 28. At the same time, the random code N is encrypted by RSA engine 16 to produce encrypted data RSA (N). Alternatively, the random code N is encrypted by DES engine 18 to produce encrypted data DES (N). The RSA engine 16 and the DES engine 18 may even be used together to encrypt data for improved reliability.

The encrypted data RSA(N) or the encrypted data DES(N) is passed back to the external device 30 through the internal bus 28 and the host bus 24. The external device 30 next decrypts the encrypted data RSA(N) or DES(N) to produce decrypted data N. Through the host bus 24 and the internal bus 28, the decrypted data N is sent back to the BIOS chip 10. The decrypted data N is compared with the random code N in the second flash memory 14. If the decrypted data N and the random code N match, data for modifying BIOS is input from the external device 30 until the modification command WE# is no longer present. On the other hand, if the decrypted data N and the random code N do no match, any modification data from the external device 30 is rejected.

The encryption of code data by the RSA engine 16 to produce the encrypted data RSA(N) or the encryption of code data by the DES engine 18 to produce the encrypted data DES(N) ensures no virus program can secure correct RSA or DES codes. Hence, any attempt by virus program to infuse incorrect data into the BIOS is bound to fail. Furthermore, the delivery of the encrypted data RSA(N) or the encrypted data DES(N) to the external device for decryption followed by passing back the decrypted data to the code administrator for examination adds another level of security to the internal BIOS chip data.

It will be apparent to those skilled in the art that various modifications and variations can be made to the structure of the present invention without departing from the scope or spirit of the invention. In view of the foregoing, it is intended that the present invention cover modifications and variations of this invention provided they fall within the scope of the following claims and their equivalents.

4. BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawing is included to provide a further understanding of the invention, and is incorporated in and constitutes a part of this specification. The drawing illustrates embodiments of the invention and, together with the description, serves to explain the principles of the invention. In the drawing,

Fig. 1 is a sketch showing the code administration system within a BIOS chip according to one preferred embodiment of this invention.

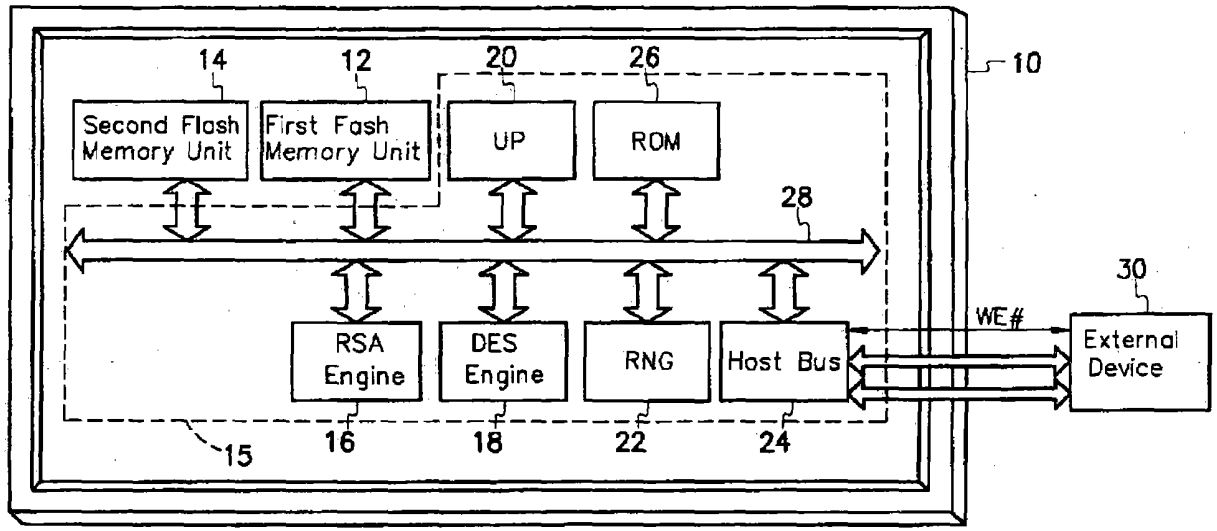

14
12
20
26
10
Second Flash Memory Unit
First Flash Memory Unit
UP
ROM
28
30
WE#
RSA Engine
DES Engine
RNG
Host Bus
External Device
15
16
18
22
24

FIG. 1

ABSTRACT OF THE DISCLOSURE

A BIOS chip having a code administrator therein. The BIOS chip includes a first flash memory unit for holding internal BIOS data, a second flash memory unit for holding code data and an integrated code administration device. The integrated code administration device is connected to an external device, the first flash memory and the second flash memory. After receiving a modification command, the integrated code administration device generates code data and transmits to the second flash memory unit for storage. A data encryption of the coded data is next carried out. The encrypted data is sent to the external device for de-encryption. Finally, the original coded data and the decrypted data is compared. Only when the original coded data and the decrypted data match each other will the permission to modify any internal BIOS data be granted. Since the said encryption can be carried out using a non-symmetrical RSA engine, correct RSA code is impossible to derive. Hence, internal BIOS data is protected against modification through any infiltration by virus programs.